Introducción a ML y GenAI

Preprocesamiento de Datos

Ariel Ramos Vela

17-09-2024

Agenda

1. Introducción al Preprocesamiento de Datos
2. Importancia del Preprocesamiento
3. Ejemplo: Titanic Dataset
4. Tipos de Preprocesamiento
    1. Limpieza de datos
    2. Manejo de datos faltantes
    3. Conversión de datos categóricos
    4. Escalado y normalización
5. Conclusiones y recomendaciones
6. Preguntas y respuestas

Introducción al Preprocesamiento de Datos

- El preprocesamiento es un paso fundamental en el desarrollo de modelos de Machine Learning.
- Objetivo: Preparar los datos para que los algoritmos puedan extraer patrones y hacer predicciones de manera efectiva.
- El preprocesamiento mejora la **calidad** y **coherencia** de los datos.

¿Por qué es Importante el Preprocesamiento?

1. **Limpieza de Datos**
    1. Eliminación de duplicados
    2. Corrección de inconsistencias
2. **Manejo de Datos Faltantes**
    1. Eliminación de filas o columnas
    2. Imputación
3. **Conversión de Datos Categóricos**
    1. One-hot encoding
    2. Label encoding
4. **Escalado y Normalización**
    1. Escalado Min-Max
    2. Normalización Z-score

Introducción al Titanic Dataset

- **Titanic Dataset** es uno de los conjuntos de datos más utilizados para **aprendizaje supervisado**.
- Contiene información sobre los pasajeros del Titanic, que se hundió en 1912.

- **Objetivo**: Predecir si un pasajero sobrevivió o no, basado en las características disponibles.

Features Principales:

- **PassengerId**: Identificador único del pasajero.
- **Pclass**: Clase del boleto (1ª, 2ª, 3ª).
- **Name**: Nombre del pasajero.
- **Sex**: Género (masculino/femenino).
- **Age**: Edad del pasajero.
- **SibSp**: Número de hermanos/esposos a bordo.
- **Parch**: Número de padres/hijos a bordo.
- **Fare**: Precio del boleto.
- **Cabin**: Número de cabina (si disponible).
- **Embarked**: Puerto donde embarcó el pasajero (C = Cherburgo, Q = Queenstown, S = Southampton).
- **Survived**: Indicador binario de supervivencia (0 = No, 1 = Sí).

- **Tamaño del Dataset:**
  - **Total de registros**: 891 pasajeros
  - **Datos tabulares**: Organizados en filas y columnas.

Datos Categóricos vs Datos Numéricos

PassengerId	Pclass	Name	Sex	Age	Fare	Survived
1	3	Braund, Mr. Owen	male	22	7.25	0
2	1	Cumings, Mrs. John	female	38	71.28	1
3	3	Heikkinen, Miss. Laina	female	26	7.92	1

Datos Categóricos

- Representan **categorías o etiquetas**.
- **Ejemplo**:
  - **Sex**: "male" y "female" (género).
  - **Pclass**: 1, 2, 3 (clase del boleto).
  - **Survived**: 0 (no sobrevivió) o 1 (sobrevivió).
- Generalmente no tienen un **orden** inherente entre sus valores.

Datos Numéricos

- Representan **valores cuantitativos**.
- **Ejemplo**:
  - **Age**: 22, 38, 26 (edad en años).
  - **Fare**: 7.25, 71.28, 7.92 (precio del boleto).
- Pueden ser utilizados directamente en cálculos matemáticos.

¿Qué representan las columnas y las filas?

PassengerId	Pclass	Name	Sex	Age	Fare	Survived
1	3	Braund, Mr. Owen	male	22	7.25	0
2	1	Cumings, Mrs. John	female	38	71.28	1
3	3	Heikkinen, Miss. Laina	female	26	7.92	1

Columnas (Features)

- Representan **atributos** o **características** de los datos.
- Cada columna contiene un **tipo de información específica** sobre las observaciones (filas).
- En Machine Learning, las columnas son conocidas como **features**.
- **Ejemplos en el Titanic Dataset**:
  - **Age** (Edad): La edad del pasajero.
  - **Sex** (Sexo): El género del pasajero.
  - **Fare** (Tarifa): El precio del boleto.
  - **Pclass** (Clase): La clase del boleto (1ª, 2ª, 3ª).

Filas (Rows)

- Cada fila es una **observación individual** o un **registro**.
- Representa un **ejemplo específico** en el conjunto de datos.
- En el Titanic Dataset, cada fila corresponde a un **pasajero**.
- **Ejemplo en el Titanic Dataset**:
  - Fila 1: Pasajero masculino de 22 años, en clase 3, con tarifa de $7.25.

Tipos de Preprocesamiento

1. **Limpieza de Datos**
    1. Eliminación de duplicados
    2. Corrección de inconsistencias
2. **Manejo de Datos Faltantes**
    1. Eliminación de filas o columnas
    2. Imputación
3. **Conversión de Datos Categóricos**
    1. One-hot encoding
    2. Label encoding
4. **Escalado y Normalización**
    1. Escalado Min-Max
    2. Normalización Z-score

Limpieza de Datos

- **Detección de datos duplicados**: Ejemplo, dos pasajeros con el mismo nombre y detalles en el Titanic Dataset.

- **Corrección de inconsistencias**: Diferentes formatos para la misma información, como “male” y “Male”.

PassengerId	Name	Sex	Age	Pclass	Fare
1	Braund, Mr. Owen	male	22	3	7.25
2	Cumings, Mrs. John	Female	38	1	71.28
3	Heikkinen, Miss Laina	female	26	3	7.92
4	Allen, Mr. William	Male	35	1	8.05

Manejo de Datos Faltantes

- **Datos faltantes en el Titanic Dataset**:
  - Edad (Age)
  - Cabina (Cabin)
- **Estrategias:**
- **Eliminar filas/columnas**: Si el porcentaje de datos faltantes es alto.

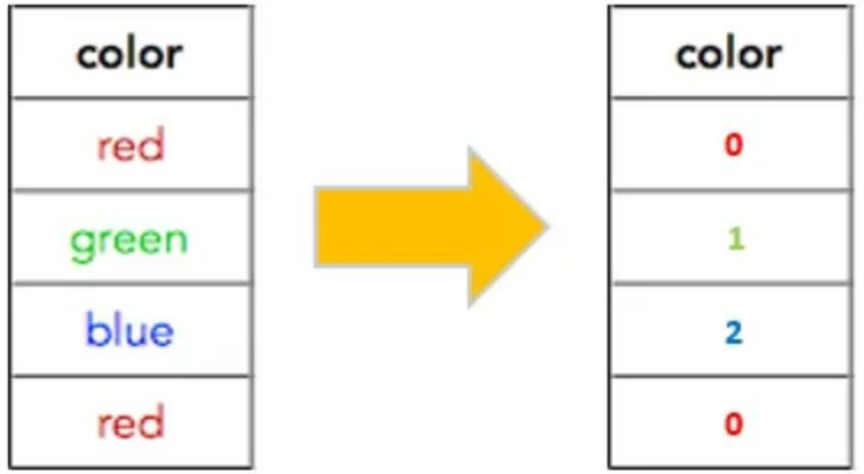
- **Imputar datos**: Usar la mediana o la media para rellenar los valores faltantes.
  - Ejemplo: Imputar la edad faltante con la mediana de las edades.

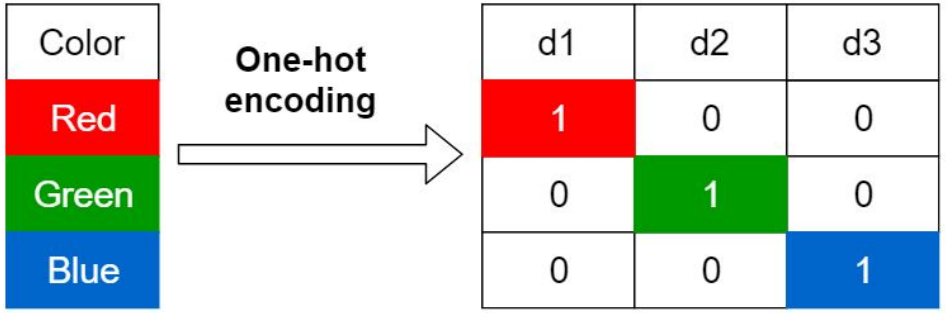
PassengerId	Name	Age	Pclass	Cabin	Embarked
1	Braund, Mr. Owen	22	3	NaN	S
2	Cumings, Mrs. John	38	1	C85	C
3	Heikkinen, Miss Laina	26	3	NaN	S
4	Allen, Mr. William	35	1	B28	S
5	Moran, Mr. James	NaN	3	NaN	NaN

Conversión de Datos Categóricos

- **Datos categóricos** en el Titanic Dataset:
  - Sexo (Sex): “male” y “female”
  - Clase del boleto (Pclass): 1, 2, 3
- **Métodos:**

1. **Label Encoding**: Convertir “male” en 0 y “female” en 1.
2. **One-Hot Encoding**: Crear columnas binarias para cada categoría de “Pclass”.





One-Hot Encoding

- **Ventajas:**
  - **Sin orden implícito**: Útil cuando no existe un orden natural entre las categorías.
  - **Evita sesgos**: Previene que el modelo asuma una jerarquía entre categorías.
- **Desventajas:**
  - **Aumenta la dimensionalidad**: Puede crear muchas columnas, especialmente en variables con muchas categorías, lo que incrementa el costo computacional.
  - **Ineficiencia**: Para conjuntos de datos grandes, puede ser ineficiente.

Label Encoding

- **Ventajas:**
  - **Simplicidad**: No aumenta la dimensionalidad del dataset.
  - **Rápido y eficiente**: Requiere menos memoria y es más eficiente para conjuntos de datos grandes.
- **Desventajas:**
  - **Orden implícito**: El modelo puede interpretar que las categorías tienen un orden o jerarquía, lo que puede introducir **sesgo** si el orden no es relevante.

Otras Técnicas de Encoding...

1. Target Encoding	**Descripción**: Reemplaza cada categoría por el promedio de la variable objetivo (target) en cada categoría.	**Uso**: Útil en problemas de clasificación, pero puede causar **overfitting** si no se maneja adecuadamente.	**Ejemplo**: En el Titanic Dataset, reemplazar "Sex" con la tasa de supervivencia media por género.
2. Frequency Encoding	**Descripción**: Cada categoría se codifica según su **frecuencia** en el dataset.	**Ventaja**: Compacta y eficiente, especialmente para datos con muchas categorías.	**Ejemplo**: Reemplazar "Embarked" con la frecuencia de cada puerto en el dataset.
3. Binary Encoding	**Descripción**: Convierte categorías en binarios de forma compacta. Cada categoría se convierte en un número entero y luego en su representación binaria.	**Ventaja**: Reduce la dimensionalidad en comparación con One-Hot Encoding.	**Ejemplo**: Categoría "Pclass" (1, 2, 3) se codifica en binario: 1 = 001, 2 = 010, 3 = 011.

Normalización y Estandarización

- Los algoritmos de Machine Learning basados en distancias (SVM, KNN) necesitan datos en la **misma escala**.
- **Métodos:**
  1. **Normalización (Min-Max Scaling)**:
  2. **Estandarización (Z-Score Scaling)**:

- **Ejemplo:**
  - Normalizar la columna **Fare** en el Titanic Dataset para que todos los precios de boletos tengan la misma escala.

Normalización (Min-Max Scaling)

- **Propósito**: Reescalar los datos para que caigan dentro de un rango específico, comúnmente entre **0 y 1**.
- **Uso**: Es útil cuando queremos que los datos se mantengan dentro de un rango fijo, como en **redes neuronales** o cuando las características tienen diferentes escalas.
- **Sensibilidad a outliers**: Muy sensible a valores atípicos (outliers), ya que estos pueden afectar drásticamente el rango mínimo y máximo.
- **Ejemplo**: Columna de **precios** que varían en gran magnitud (por ejemplo, de $0 a $1000) puede ser normalizada para ajustarse entre **0 y 1**.

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}}$$

Estandarización (Z-Score Scaling)

- **Propósito**: Reescalar los datos de manera que tengan **media 0** y **desviación estándar 1**.
- **Uso**: Útil cuando los datos tienen una **distribución normal** o en modelos como **regresión lineal, SVM, KNN**, que son sensibles a las diferentes escalas.
- **Sensibilidad a outliers**: Menos sensible que la normalización, pero aún puede verse afectada por valores atípicos, ya que influencian la media y desviación estándar.
- **Ejemplo**: Columna de **edades**, donde los valores están distribuidos en torno a una media.

$$Z = \frac{X - \mu}{\sigma}$$

- **X**: Valor de la característica.
- **μ**: Media de la característica.
- **σ**: Desviación estándar de la característica.

Conclusiones

- El preprocesamiento es crucial para mejorar la **calidad** y **rendimiento** de los modelos.
- Diferentes tipos de preprocesamiento deben aplicarse según el tipo de datos y su contexto.
- En el Titanic Dataset, las técnicas de imputación, conversión categórica y escalado mejoran la precisión del modelo.

Recomendaciones

Siempre **explorar** los datos antes de aplicar preprocesamiento.

Utilizar **gráficos** para detectar valores atípicos y anomalías.

Evaluar el impacto de cada técnica de preprocesamiento en el rendimiento del modelo.

Siguientes pasos...

- Taller: Preprocesamiento de datos.
- El material lo podrás encontrar en el Github repository del curso.